Algorithmes Gradient-Proximaux Stochastiques

Gersende Fort, Laurent Risser, Éric Moulines, Edouard Ollier, Adeline Samson

Algorithmes Gradient-Proximaux Stochastiques

Gersende FORT[1], Laurent RISSER[1], Eric MOULINES[2], Edouard OLLIER[3], Adeline SAMSON[4]

[1]IMT, Université de Toulouse et CNRS, Toulouse, France

[2]CMAP, Ecole Polytechnique, Palaiseau, France

[3]UMPA, Ecole Normale Supérieure de Lyon et CNRS, Lyon, France

[4]LJK, Université Grenoble-Alpes et CNRS, Grenoble, France

gersende.fort@math.univ-toulouse.fr, laurent.risser@math.univ-toulouse.fr,
eric.moulines@polytechnique.edu, edouard.ollier@ens-lyon.fr,
adeline.leclercq-samson@univ-grenoble-alpes.fr

Résumé – Motivés par des applications en inférence statistique, nous proposons deux versions stochastiques de l'algorithme Gradient Proximal pour la maximisation de fonctions composites. Nous établissons la convergence dans le cas concave, lorsque les approximations Monte Carlo sont biaisées avec un biais qui ne s'atténue pas au cours des itérations.

Abstract – Motivated by applications in statistical inference, we propose two versions of a Stochastic Proximal Gradient algorithm for the maximization of composite functions. We establish the convergence in the concave case, when the Monte Carlo approximation is biased and the bias does not vanish along iterations.

1 Introduction

Nous nous intéressons à des méthodes numériques pour la résolution du problème d'optimisation

$$\mathrm{argmax}_{\theta\in\mathbb{R}^d}\,(f(\theta)-g(\theta)) \tag{1}$$

lorsque les fonctions f et g vérifient :

H1 *La fonction $g:\mathbb{R}^d\to[0,+\infty]$ est convexe, non identiquement égale à $+\infty$ et semi-continue inférieurement. La fonction $f:\mathbb{R}^d\to\mathbb{R}\cup\{-\infty\}$ est de classe C^1 sur $\Theta:=\{\theta\in\mathbb{R}^d : g(\theta)+|f(\theta)|<\infty\}$ et de gradient L-Lipschitz.*

On se restreint de plus au cas où la fonction f n'a pas d'expression explicite, et son gradient s'exprime comme une intégrale qui n'a pas non plus d'expression explicite. La solution retenue est de lire cette intégrale comme une espérance, et utiliser des méthodes d'optimisation du premier ordre couplées avec des méthodes de Monte Carlo (MC). Ce contexte est suggéré par un certain nombre d'applications.

Comme premier exemple, citons la résolution de problèmes d'inférence statistique dans les modèles à données cachées : la fonction f est alors la log-vraisemblance des observations et elle s'écrit à l'aide de la marginale de la vraisemblance des données manquantes et des observations (la dépendance en les observations est omise)

$$f(\theta)=\log\int_{\mathsf{Z}}p(z,\theta)\mu(\mathrm{d}z).$$

Sous des conditions de régularité du modèle, la fonction f est dérivable de gradient donné par

$$\theta\mapsto\int_{\mathsf{Z}}\partial_\theta\log p(z,\theta)\,\pi_\theta(z)\mu(\mathrm{d}z),$$

avec $\pi_\theta(z):=p(z,\theta)\exp(-f(\theta))$. Puisque f est non explicite, la loi d'intégration $\pi_\theta\mathrm{d}\mu$ est connue à une constante de normalisation près et les techniques d'approximation Monte Carlo efficaces devront utiliser des échantillonneurs de Monte Carlo par Chaînes de Markov (MCMC) [17]- les méthodes d'échantillonnage d'importance sont en effet particulièrement peu recommandées lorsque $\mathsf{Z}\subseteq\mathbb{R}^s$ avec s plus grand que la dizaine. Parmi les modèles à données cachées, le cas dit modèle exponentiel est très répandu : $\log p$ est de la forme $\phi(\theta)+\langle S(z),\psi(\theta)\rangle$ de sorte que le gradient de f s'écrit

$$\nabla f(\theta)=\nabla\phi(\theta)+J_\psi(\theta)\int_{\mathsf{Z}}S(z)\,\pi_\theta(z)\,\mathrm{d}\mu(z),$$

où J_ψ désigne la matrice jacobienne associée à ψ, ∇ l'opérateur de gradient et $\langle\cdot,\cdot\rangle$ le produit scalaire usuel sur $\mathbb{R}^q$.

Un second exemple est donné par l'inférence de paramètres d'une mesure de Gibbs, problème rencontré notamment dans l'inférence d'un modèle de réseau dont les noeuds sont à valeur dans un alphabet fini. Dans ce contexte, on dispose de N observations $(Y^{(1)},\cdots,Y^{(N)})$, supposées indépendantes et identiquement distribuées selon une mesure de Gibbs proportionnelle à $\exp(\langle S(y),\theta\rangle)$, où $Y^{(i)}\in\{1,\cdots,M\}^p$ collecte les valeurs des p noeuds du graphe. La log-vraisemblance f est de

la forme

$$\theta \mapsto \sum_{k=1}^{N} \left\langle S(Y^{(k)}), \theta \right\rangle - N \log \int_{\mathsf{Z}} \exp\left(\langle S(z), \theta \rangle\right) \mathrm{d}\mu(z),$$

où le dernier terme, connu sous le nom de fonction de partition, est non explicite. Sous des conditions de régularité du modèle, le gradient ∇f est donné par

$$\nabla f(\theta) = \sum_{k=1}^{N} S(Y^{(k)}) - N \int_{\mathsf{Z}} S(z) \pi_\theta(z) \mathrm{d}\mu(z)$$

avec $\pi_\theta(z) \propto \exp(\langle S(z), \theta \rangle)$. Là encore, π_θ étant connue à une constante de normalisation près, l'approximation stochastique du gradient ne pourra qu'utiliser des méthodes MCMC.

Dans les applications considérées, g traduira une contrainte sur le paramètre θ telle qu'une contrainte de parcimonie (e.g. lorsque $g(\theta) = \lambda \sum_{i=1}^{d} |\theta_i|$ avec $\lambda > 0$), ou une contrainte d'appartenance à un sous-ensemble convexe fermé $\mathcal{K}$ de $\mathbb{R}^d$ (e.g. $g(\theta) = 0$ si $\theta \in \mathcal{K}$ et $+\infty$ sinon), ou une somme de telles contraintes.

Il existe dans la littérature un certain nombre de contributions sur des versions stochastiques d'algorithmes d'optimisation de premier ordre, pour la maximisation de fonctions composites (voir par exemple [4, 5, 6, 8, 9, 14] et les références citées). Néanmoins, à quelques rares exceptions dont [8], les algorithmes sont étudiés sous l'hypothèse d'approximations MC sans biais, ce qui n'est pas vrai pour des approximations MCMC : si $\{Z_j, j \geq 0\}$ est une chaîne de Markov ergodique d'unique loi invariante $\pi \mathrm{d}\mu$, alors

$$\lim_{m \to \infty} \mathbb{E}[m^{-1} \sum_{j=1}^{m} S(Z_j)] = \int S(z) \pi(z) \mathrm{d}\mu(z),$$

mais l'espérance à m fixe n'est jamais égale à la limite. Autrement dit, l'approximation MCMC est biaisée mais on peut rendre ce biais arbitrairement petit en augmentant le nombre de points m de l'approximation Monte Carlo. Néanmoins, pour des questions de coût de calcul dans des méthodes itératives qui contiendront une telle approximation à chaque itération, on souhaite garder fixe le nombre de tirages m au cours des itérations. Il est donc nécessaire de savoir établir le bien-fondé d'algorithmes d'optimisation stochastiques, reposant sur des approximations biaisées avec un biais qui ne s'évanouit pas au cours des itérations : là est une difficulté à notre connaissance non couverte par la littérature. Nous apportons une réponse dans le cas d'une version stochastique de l'algorithme gradient-proximal.

La seconde originalité de notre contribution est de fournir une analyse de convergence sans supposer que la fonction f est fortement concave, cette condition n'étant généralement pas vérifiée pour les deux exemples cités ci-dessus. Nous donnons des résultats dans le cas où f est concave, ce qui convient à l'exemple d'inférence dans un modèle de Gibbs mais reste insuffisant pour le premier exemple. L'analyse de convergence dans le cas f non concave est, à notre connaissance, un problème ouvert lorsque l'approximation MC est biaisée (voir [13] par exemple pour le cas non biaisé).

2 Algorithmes MCPG et SAPG

Pour la résolution du problème (1), nous proposons un algorithme itératif de type Minoration-Majoration stochastique, tel que chaque itération combine une étape de gradient stochastique basée sur des algorithmes MCMC, et une étape de proximal que nous supposerons explicite (voir [2] pour des résultats dans le cas où cette étape est approchée). Plus précisément, chaque itération est une version bruitée de la dynamique Gradient-Proximal (voir [7])

$$\tau_{n+1} = \mathrm{Prox}_{\gamma_{n+1}, g}\left(\tau_n + \gamma_{n+1} \nabla f(\theta_n)\right) \tag{2}$$

où $\{\gamma_n, n \geq 0\}$ est une suite de pas déterministe choisie par l'utilisateur, et l'opérateur $\mathrm{Prox}_{\gamma, g}$ est défini par (voir [16])

$$\mathrm{Prox}_{\gamma, g}(\tau) := \mathrm{argmin}_{\theta \in \Theta} \left(g(\theta) + \frac{1}{2\gamma} \|\theta - \tau\|^2 \right).$$

Compte-tenu des applications visées décrites en Section 1, nous présentons cet algorithme dans le cas où ∇f vérifie la condition suivante

H2 $\nabla f(\theta) = \Psi(\theta)\, \bar{S}(\theta)$ *où* $\Psi : \Theta \to \mathbb{R}^{d \times q}$*, et* $\overline{S}(\theta)$ *est l'espérance d'une fonction* $S : \mathsf{Z} \to \mathbb{R}^q$ *par rapport à une mesure de probabilité* $\pi_\theta \, \mathrm{d}\mu$ *sur* Z *:*

$$\overline{S}(\theta) = \int_{\mathsf{Z}} S(z)\, \pi_\theta(z)\, \mathrm{d}\mu(z).$$

On trouvera dans [1] des algorithmes Gradient-Proximaux perturbés valables pour des formes générales de ∇f, ainsi qu'une analyse de convergence de ces schémas itératifs bruités y compris dans le cas de perturbations stochastiques biaisées.

Les algorithmes. Etant donné une suite déterministe $\{\gamma_n, n \geq 0\}$ à valeur dans $]0, 1/L]$, une suite déterministe $\{m_n, n \geq 0\}$ à valeur dans $\mathbb{N}$, et une valeur initiale $\theta_0 \in \Theta$, on considère l'algorithme *Stochastic Proximal-Gradient* donné par

$$\theta_{n+1} = \mathrm{Prox}_{\gamma_{n+1}, g}\left(\theta_n + \gamma_{n+1} \Psi(\theta_n) S_{n+1}\right) \tag{3}$$

où S_{n+1} est une approximation Monte Carlo de la quantité inconnue $\overline{S}(\theta_n)$, basée sur une chaîne de Markov $\{Z_{j,n}, j \leq m_{n+1}\}$ de noyau de transition P_{θ_n}, et d'unique mesure invariante $\pi_{\theta_n} \mathrm{d}\mu$. Deux schémas d'approximation et donc deux algorithmes sont proposés. Tout d'abord, l'algorithme *Monte Carlo Proximal-Gradient* (MCPG), où pour tout n, on pose $S_{n+1} = S^{\mathsf{mc}}_{n+1}$ avec

$$S^{\mathsf{mc}}_{n+1} := m^{-1}_{n+1} \sum_{j=1}^{m_{n+1}} S(Z_{j,n})\,;$$

il repose sur une approximation de $\overline{S}(\theta_n)$ basée uniquement sur les tirages courant $\{Z_{j,n}, j \geq 1\}$. Ensuite, l'algorithme *Stochastic Approximation Proximal-Gradient* (SAPG), où pour tout n, on pose $S_{n+1} = S^{\mathsf{sa}}_{n+1}$ avec

$$S^{\mathsf{sa}}_{n+1} := (1 - \delta_{n+1}) S^{\mathsf{sa}}_n + \delta_{n+1}\, m^{-1}_{n+1} \sum_{j=1}^{m_{n+1}} S(Z_{j,n}).$$

Ici, $\{\delta_n, n \geq 0\}$ désigne une suite déterministe à valeur dans $]0,1[$ choisie par l'utilisateur. SAPG repose sur une approximation lissée de $\overline{S}(\theta_n)$ qui, de par sa forme récursive, est une combinaison linéaire des images par S de l'ensemble des tirages effectués depuis l'initialisation de l'algorithme.

Pour les deux algorithmes, le nombre de tirages Monte Carlo à la fin de l'itération n est $\sum_{i=1}^{n} m_i$. Néanmoins, vu la forme récursive de S_{n+1} dans SAPG, il est attendu que ce dernier ait un meilleur comportement. Nous donnons deux résultats pour appuyer cette intuition : la proposition 1 qui donne une vitesse de convergence de $\mathbb{E}\left[\|S_{n+1} - \overline{S}(\theta_n)\|^2\right]$ pour les deux algorithmes, et une analyse numérique (voir Section 3 ; voir aussi [12] pour une application à l'inférence pénalisée dans des modèles à données cachées).

Dans le cas où f est la log-vraisemblance dans un modèle à données cachées, on peut montrer (voir [12]) que (2) est un algorithme *Generalized Expectation-Maximization* (GEM), de sorte que (3) correspond à un algorithme *Stochastic GEM*. Les algorithmes MCPG et SAPG sont directement inspirés des approximations stochastiques faites dans les algorithmes MCEM et SAEM respectivement proposés par [18] et [10].

Comportement asymptotique. Nous donnons un ensemble de conditions suffisantes pour établir la convergence presque-sûre des trajectoires de MCPG et de celles de SAPG vers une solution de (1). Pour ce faire, nous nous restreignons au cas où la fonction f est concave, de sorte que $f - g$ l'est aussi. Nous supposons aussi que les algorithmes MCMC vérifient des conditions d'ergodicité, afin de garantir le bien-fondé de l'approximation de $\overline{S}(\theta)$ par une somme de Monte Carlo. Puisque le long d'une trajectoire de l'algorithme, les échantillons Monte Carlo sont simulés par des noyaux différents $P_{\theta_1}, P_{\theta_2}, \cdots$ où $\{\theta_n, n \geq 0\}$ est une suite aléatoire qui dépend de tout le passé de l'algorithme, nous devons formuler des conditions de régularité sur $\mathrm{Prox}_{\gamma,g}$, sur les noyaux markoviens $\{P_\theta, \theta \in \Theta\}$ et sur les densités $\{\pi_\theta, \theta \in \Theta\}$ afin de garantir que "tous les noyaux se ressemblent" lorsque la suite $\{\theta_n, n \geq 0\}$ converge. Enfin, nous imposons des conditions sur les pas $\{\gamma_n, n \geq 0\}$, $\{\delta_n, n \geq 0\}$ et la taille de la somme de Monte Carlo $\{m_n, n \geq 0\}$ - nous donnons ici ces conditions pour des formes particulières de ces suites (voir [1, 12] pour le cas plus général). Plus rigoureusement, nous établissons les résultats de convergence sous les conditions suivantes :

H3 *f est concave, et l'ensemble $\mathcal{L} := \mathrm{argmax}_\Theta(f - g)$ est non vide. Θ est borné.*

H4 *Il existe une constante L telle que pour tout $\theta, \theta' \in \Theta$, $\|\Psi(\theta) - \Psi(\theta')\| + \|\overline{S}(\theta) - \overline{S}(\theta')\| \leq L\|\theta - \theta'\|$. De plus, $\sup_{\gamma\in]0,1/L]} \sup_{\theta\in\Theta} \gamma^{-1} \|\mathrm{Prox}_{\gamma,g}(\theta) - \theta\| < \infty$.*

Ici, $\|\cdot\|$ désigne la norme euclidienne. On rappelle (voir e.g. [15]) que pour des fonctions mesurables $f : \mathsf{Z} \to \mathbb{R}$ et $V : \mathsf{Z} \to [1, +\infty[$, $|f|_V$ désigne $\sup_{\mathsf{Z}} |f|/V$; et pour une mesure signée ν sur Z, $\|\nu\|_V$ désigne $\sup_{f:|f|_V\leq 1} |\nu(f)|$.

H5 *Il existe $\lambda \in [0,1[$, $b < \infty$ et une fonction mesurable $W : \mathsf{Z} \to [1, +\infty[$ tels que $|S|_{\sqrt{W}} < \infty$ et $\sup_{\theta\in\Theta} P_\theta W \leq \lambda W + b$.*

Il existe des constantes $C < \infty$ et $\rho \in]0,1[$ telles que pour tout $z \in \mathsf{Z}$ et $n \geq 0$,

$$\sup_{\theta\in\Theta} \|P_\theta^n(z,\cdot) - \pi_\theta\|_W \leq C\,\rho^n\,W(z).$$

Il existe une constante C t.q. pour tout $\theta, \theta' \in \Theta$,

$$\|\pi_\theta - \pi_{\theta'}\|_{\sqrt{W}} + \sup_{z\in\mathsf{Z}} \frac{\|P_\theta(z,\cdot) - P_{\theta'}(z,\cdot)\|_{\sqrt{W}}}{\sqrt{W}(z)} \leq C\,\|\theta - \theta'\|.$$

On trouvera dans [1] des indications pour vérifier la condition H5, et des conditions suffisantes sur g entrainant la condition de régularité de $\mathrm{Prox}_{\gamma,g}$ énoncée dans H4 (les pénalités *elastic-net*, *Lasso*, et *fused Lasso* vérifient la condition lorsque Θ est un compact convexe).

H6 *$\gamma_n = \gamma_\star n^{-a}$, $\delta_n = \delta_\star n^{-b}$ et $m_n = \lceil m_\star n^c \rceil$ avec $\gamma_\star \in]0, 1/L]$, $\delta_\star \in]0,1[$, $m_\star > 0$ et $c \geq 0$, $0 \leq b \leq a \leq 1$.*

Nous affirmons (voir [11, Proposition 2] et [12, Proposition 5])

Proposition 1 *Sous les conditions H1 à H6,*

$$\mathbb{E}\left[\|S_{n+1}^{\mathsf{mc}} - \overline{S}(\theta_n)\|^2\right] = O\left(n^{-c}\right),$$
$$\mathbb{E}\left[\|S_{n+1}^{\mathsf{sa}} - \overline{S}(\theta_n)\|^2\right] = O\left(n^{-2((a-b)\wedge(b+c)/2)}\right).$$

En particulier, lorsque le nombre de tirages Monte Carlo n'augmente pas au cours des itérations ($c = 0$), l'erreur $S_{n+1}^{\mathsf{mc}} - \overline{S}(\theta_n)$ ne tend pas vers zero ; l'erreur $S_{n+1}^{\mathsf{sa}} - \overline{S}(\theta_n)$ tend vers zero dès que $a > b > 0$, même avec $c = 0$. Pour MCPG et SAPG, nous prouvons le résultat de convergence suivant (voir [1, Théorèmes 4 et 6] et [12, Théorème 6])

Théorème 2 *Sous les conditions H1 à H6 et*

$$(a \in]1/2, 1], c = 0) \quad \textit{ou} \quad (a \in [0,1], c > 1 - a), \qquad (4)$$

il existe une v.a. θ_∞ à valeur dans $\mathcal{L}$ telle que avec probabilité un, $\lim_n \theta_n = \theta_\infty$ où $\{\theta_n, n \geq 0\}$ est la suite donnée par MCPG. On a le même résultat pour SAPG en remplaçant la condition (4) *par $(a \in]1/2, 1], b \in [0, 2a - 1[, c = 0)$.*

Il faut noter la convergence de MCPG même lorsque $c = 0$ i.e. même lorsque l'erreur $S_{n+1}^{\mathsf{mc}} - \overline{S}(\theta_n)$ ne s'annule pas asymptotiquement ; il faut néanmoins compenser cette persistance du bruit par des pas $\{\gamma_n, n \geq 0\}$ décroissants ($a \in]1/2, 1]$) : on retrouve le régime de l'Approximation Stochastique [3]. Pour SAPG, on ne reporte pas les conditions lorsque $c > 0$, car en pratique, cette solution n'est jamais mise en oeuvre.

Nos hypothèses permettent de traiter le cas d'une approximation Monte Carlo biaisée ; lorsqu'elle est non biaisée, certaines des conditions données dans H4 et H5 sont omises (voir [1, 12]).

3 Estimation des paramètres d'une mesure de Gibbs

Puisque nos résultats théoriques concernent le cas concave, nous concluons ce papier par un exemple numérique qui vérifie cette condition. Nous reprenons le modèle de graphe présenté en section 1 avec $M = 5$ et $p = 100$; le modèle de Gibbs comprend un terme d'interactions possiblement entre toute paire de noeuds, et un champ externe, ce que l'on modélise par

$$\langle S(y), \theta \rangle = \sum_{i=1}^{p} \theta_{ii} S_{ii}(y) + \sum_{i=1}^{p-1} \sum_{j=i+1}^{p} \theta_{ij} S_{ij}(y)$$

avec $S_{ii}(y) = y_i$ et $S_{ij}(y) = \mathbb{1}_{y_i = y_j}$ pour $i < j$. La pénalité g porte sur la norme L_1 et la positivité des éléments $\{\theta_{ij}, i \leq j\}$. Les $N = 250$ observations sont simulées comme la réalisation $Z_{J,\star}$ d'un MCMC à l'itération $J = 500$, de mesure cible la mesure de Gibbs pour un paramètre $\theta_\star \in \mathbb{R}^{p(p+1)/2}$ contenant environ 200 valeurs non nulles ; nous utilisons l'échantillonneur de Wolff [19]. SAPG et MCPG sont implémentés avec $a = 2/3$, $b = 1/3$, $m_n = 50$ (γ_n est constant pendant quelques itérations).

La figure 1[haut] montre l'évolution de la norme L_1, $n \mapsto \|\theta_n\|_{L_1}$ pour SAPG (trait plein) et MCPG (trait pointillé). La figure 1[bas] représente le boxplot de la norme de θ_n calculé sur 50 réalisations indépendantes de chaque algorithme ; $n \in \{50, 100, 150, 200, 250, 300\}$.

Ces graphes illustrent la convergence plus rapide et la variabilité plus faible de SAPG. Néanmoins, la mise en oeuvre de SAPG est plus délicate que celle de MCPG puisqu'elle repose sur plus de paramètres d'implémentation (choix des pasq $\{\delta_n, n \geq 0\}$).

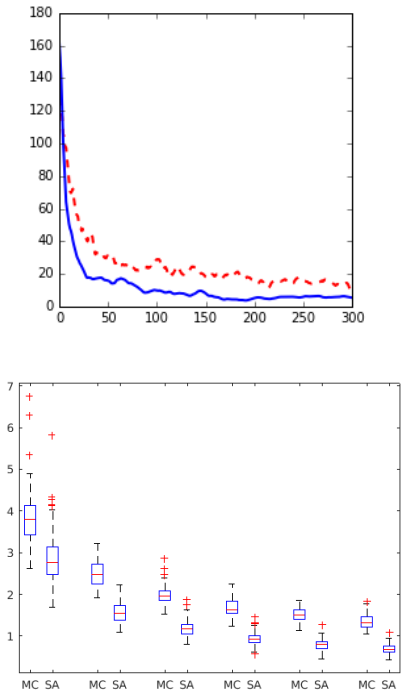

FIGURE 1 – Pour MCPG (MC) et SAPG (SA) : [haut] une trajectoire $n \mapsto \|\theta_n\|_{L_1}$; [bas] boxplot de $\|\theta_n\|$ pour différentes valeurs de n.

Références

[1] Y. Atchadé, G. Fort, and E. Moulines. On perturbed proximal gradient algorithms. *JMLR*, 18 :1–33, 2017.

[2] J.F. Aujol and C. Dossal. Stability of over-relaxations for the Forward-Backward algorithm, application to FISTA. *SIAM J. Optim.*, 25(4) :2408–2433, 2015.

[3] A. Benveniste, M. Métivier, and P. Priouret. *Adaptive algorithms and stochastic approximations*, volume 22 of *Applications of Mathematics (New York)*. Springer-Verlag, Berlin, 1990.

[4] L. Bottou, F.E. Curtis, and J. Nocedal. Optimization Methods for Large-Scale Machine Learning. Technical report, arXiv :1606 :04838, 2016.

[5] E. Chouzenoux and J.C. Pesquet. Stochastic Majorize-Minimize Subspace Algorithm for Onl ine Penalized Least Squares Estimation. Technical report, arXiv :1512 :08722, 2015.

[6] P. L. Combettes and V.R. Wajs. Signal recovery by proximal forward-backward splitting. *Multiscale Model. Simul.*, 4(4) :1168–1200, 2005.

[7] P.L. Combettes and J.C. Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, volume 49 of *Springer Optim. Appl.*, pages 185–212. Springer, New York, 2011.

[8] P.L. Combettes and J.C. Pesquet. Stochastic Approximations and Perturbations in Forward-Backward Splitting for Monotone Operators. *Online journal Pure and Applied Functional Analysis*, 1(1) :1–37, 2016.

[9] C. D. Dang and G. Lan. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. *SIAM J. Optim.*, 25(2) :856–881, 2015.

[10] B. Delyon, M. Lavielle, and E. Moulines. Convergence of a stochastic approximation version of the em algorithm. *Ann. Statist.*, 27(1) :94–128, 1999.

[11] G. Fort and E. Moulines. Convergence of the Monte Carlo expectation maximization for curved exponential families. *Ann. Statist.*, 31(4) :1220–1259, 2003.

[12] G. Fort, E. Ollier, and A. Samson. Stochastic Proximal Gradient algorithms for Penalized Mixed Models. Technical report, ArXiv 1704.08891, 2017.

[13] S. Ghadimi, G. Lan, and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Math. Program.*, 155(1-2, Ser. A) :267–305, 2016.

[14] G. Lan. An optimal method for stochastic composite optimization. *Math. Program.*, 133(1-2, Ser. A) :365–397, 2012.

[15] S. Meyn and R. L. Tweedie. *Markov chains and stochastic stability*. Cambridge University Press, Cambridge, second edition, 2009. With a prologue by Peter W. Glynn.

[16] J. Moreau. Fonctions convexes duales et points proximaux dans un espace hilbertien. *CR Acad. Sci. Paris Sér. A Math*, 255 :2897–2899, 1962.

[17] C.P. Robert. *Méthodes de Monte Carlo par chaînes de Markov*. Statistique Mathématique et Probabilité. Éditions Économica, Paris, 1996.

[18] G. Wei and M. Tanner. A Monte-Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *J. Amer. Statist. Assoc.*, 85 :699–704, 1990.

[19] U. Wolff. Collective Monte Carlo updating for spin systems. *Phys. Rev. Lett.*, 62 :361–364, 1989.